REMARKS

Claims 1-48 are pending in the present application. Reconsideration of the claims is respectfully requested.

I. 35 U.S.C. § 103, Alleged Obviousness Based on Owens

The Office Action rejects claims 1, 11-12, 20, 27, 35 and 46 under 35 U.S.C. § 103(a) as unpatentable over Owens et al. (U.S. Patent No. 6,047,284). This rejection is respectfully traversed.

With respect to independent claims 1 and 12, the Office Action states:

> With respect to claim 1, Owens discloses a method of deleting object data from a relational database, comprising: determining a structure of the relational database (col. 12, lines 3-5); determining a delete action based on the structure of the relational database (col. 11, lines 37-43 and lines 54-65); generating database modification commands based on the determined delete action (col. 11, lines 54-65 and col. 12, lines 1-20); and sending the database modification commands to a relational database server, wherein the relational database server deletes the object data from the relational database based on the database modification commands (see fig. 6 and fig. 12, col. 1, lines 12-25, col. 2, lines 22-32, col. 5, lines 38-59, col. 8, lines 18-30 and col. 11, lines 54-65).
>
> Owens does not clearly disclose, "generating database modification commands based on the determined delete action." But, however, Owens shows the object server generates the appropriate SQL calls to delete the data in the relational database (col. 11, lines 55-60). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to employ the teachings of Owens such as relational database structure, delete action and database modification commands so as to obtain a method of deleting object data from a relational database (col. 11, lines 37-65 and see fig. 6 and fig. 12) in the deletion of object from an object-relational system in a customizable and database independent manner environment...
>
> ...Claim 12 is essentially the same as claim 1 except that it is directed to a system rather than a method (data processor: fig 2, item 51, col. 4, lines 55-67 and col. 5, lines 1-10; col. 12, lines 3-5; col. 11, lines 37-43 and lines 54-65; col. 11, lines 54-65 and col. 12, lines 1-20; see fig. 6 and fig. 12, col. 1, lines 12-25, col. 2, lines 22-32, col. 5, lines 38-59,

col. 8, lines 18-30 and col. 11, lines 54-65), and is rejected for the same reason as applied to the claim 1 hereinabove.

Claim 1, which is representative of claim 12 with respect to similarly recited subject matter, reads as follows:

> 1.   A method of deleting object data from a relational database, comprising:
>   determining a structure of the relational database;
>   determining a delete action based on the structure of the relational database;
>   generating database modification commands based on the determined delete action; and
>   sending the database modification commands to a relational database server, wherein the relational database server deletes the object data from the relational database based on the database modification commands. (emphasis added)

Owens does not teach or suggest the features emphasized above. Owens teaches a method and apparatus for object oriented storage and retrieval of data from a relational database. In the method and apparatus of Owens, a contain object is instantiated by an object-oriented application and the identification of the object that is to be deleted is inserted into the container object. The application then issues an API call to the object server instructing it to delete the object identified by the container object. The object server verifies that the identifier in the container object specifies a valid object. After verification, the object server generates the appropriate SQL calls to delete the data in the relational database (see Figure 12; column 11, lines 43-57).

Owens does not teach or suggest determining a structure of a relational database, determining a delete action based on the structure of the relational database, or generating database modification commands based on the determined delete action. Moreover, there is nothing in Owens that teaches or suggests performing any of these operations in the delete operation explicitly taught in Owens. Rather, the delete operation shown in Figure 12, which is the only delete operation discussed in Owens, is not concerned with determining the structure of the relational database, determining a delete operation based on the structure of the relational database, or generation modification commands based on the determined delete action because the methodology of Owens does not recognize

that the delete operations for a relational database may need to be different based on the structure of the relational database.

The Office Action alleges that Owens teaches determining a structure of a relational database at column 12, lines 3-5 which reads "Before describing the figure, it may be beneficial to review some fundamentals of relational databases." It is not at all clear how such a statement can be interpreted to teach a step of a method in which a structure of a relational database is determined. There is nothing in this statement or any other statement of Owens that teaches or suggests determining the structure of a relational database.

Because Owens does not teach or suggest determining a structure of a relational database, Owens cannot be found to teach or suggest determining a delete operation based on the structure of the relational database. The Office Action alleges that this feature is taught by Owens at column 11, lines 37-43 and lines 54-65 which read as follows:

> There are numerous API calls that may be issued to the object server. A detailed description of all API calls is not necessary to understand the invention. However, it may be beneficial to illustrate a process of utilizing one of the available API calls. The API call that will be discussed deletes an object from the relational database. (column 11, lines 37-43)

> The object server then verifies that the Id in the container object specifies a valid object at step 557. After verification, the object server generates the appropriate SQL calls to delete the data in the relational database at step 559. In a preferred embodiment, the multiple SQL calls that may be necessary to delete an object from the relational database tables are performed as one atomic operation. In other words, the multiple SQL calls are performed as one operation conceptually so the user is not able to access a partially deleted object in the relational database. The one or more SQL calls to delete the object are received by the RDBMS and executed to perform the deletion. (column 11, lines 54-65).

There is nothing in these sections that teaches or suggests determining a delete operation based on a determined structure of a relational database. All that these sections of Owens teach or suggest is one API call that can be issued to an object server is a delete API call and that the delete API call includes verification of the identification of the object in the container object and generation of SQL calls to delete data in the relational

database. There is not even the mention of determining a structure of a relational database or determining a delete operation based on a structure of the relational database.

In addition, Owens does not teach or suggest generating database modification commands based on the determined delete action. While Owens teaches the generation of SQL calls to perform deletion of data from a relational database, the SQL calls in Owens are not based on a delete action that is determined based on a structure of a relational database. The Office Action alleges that this feature is taught by Owens at column 11, lines 54-65 (reproduced above) and column 12, lines 1-20 (although the Office Action goes on to admit that Owens in actuality does not teach this feature; see page 2 of the Office Action, last paragraph). The section of column 11, lines 54-65 merely teaches the SQL calls are generated to perform deletion of data in a relational database and the SQL calls are performed as atomic operations. The section column 12, lines 1-20 merely teaches an example of a SQL query command. There is nothing in either one of these sections, or any other section of Owens, that teaches or suggests generating database modification commands based on a delete action determined from a determined structure of a relational database.

As noted above, the Office Action in one instance states that Owens teaches generating database modification commands based on a delete action and then in another instance admits that Owens does not teach this feature but alleges that this feature is obvious. It should be noted that the Office Action's alleged motivation for modifying Owens to include generating database modification commands is to obtain a method of deleting object data from a relational database "...in a customizable and database independent manner environment" which is taken directly from page 1, lines 10-11 of the present specification and is not taken from the Owens reference, despite the allegation made by the Office Action. The alleged motivation for modifying Owens is not based on the actual teachings of Owens but is an attempt to recreate Applicant's claimed invention using Applicant's own disclosure as a guide. This is clear in that the very motivation provided is taken from Applicant's disclosure and is not found anywhere in the Owens reference. This is impermissible hindsight reconstruction using Applicant's own disclosure as a guide.

Furthermore, it would not have been obvious to make the many modifications necessary to generate Applicant's claimed invention from the teachings of Owens. There is no suggestion in Owens to modify the deletion API call discussed in Owens to include a step of determining a structure of a relational database, a step of determining a delete operation based on the determined structure, or generating modification commands based on the determined delete operation. Since there is no suggestion in Owens to include any of these steps, the only possible source of such a suggestion is Applicant's own disclosure.

With regard to claims 20, 27 and 35, Owens does not teach or suggest determining a structure of a relational database, determining one or more delete actions based on the structure of the relational database, and generating the class object based on the determined structure and the determined one or more delete actions. The determining steps have been discussed above with regard to similar features in independent claims 1 and 12. The Office Action gives a long series of citations to sections of Owens that supposedly teach the determining steps and the generating step of claim 20, and similar features in claims 27 and 35. However, Applicant has reviewed each of these sections and, while these sections discuss objects, there is no teaching or suggestion in these or any other sections of Owen to generate a class object based on a determined structure of a relational database and one or more determined delete actions. Thus, Owens does not teach any of the features of claims 20, 27 and 35.

Regarding claim 46, Owens does not teach or suggest determining a structure of a relational database, determine one or more default delete actions based on the structure of the relational database, receiving user input to modify the one or more default delete actions, or generating a class object based on the determined structure, the determined one or more delete actions and the user input. The step of determining a structure of the relational database and determining one or more delete actions based on the structure has been discussed above. The Office Action again alleges that Owens teaches the receiving and generating steps and cites a long list of sections of the Owens reference. However, upon review of these sections, it is clear that none of these sections even mention receiving a user input that modifies one or more default delete actions, let alone generating a class object based on the one or more default delete actions, the structure of

the relational database, and the user input. There simply is no teaching or suggestion in Owens regarding any of the features of claim 46.

In view of the above, Applicant respectfully submits that Owens does not teach or suggest the features of independent claims 1, 12, 20, 27, 35 and 46. At least by virtue of its dependency on claim 1, Owens does not teach or suggest the features of dependent claim 11. Accordingly, Applicant respectfully requests withdrawal of the rejection of claims 1, 11-12, 20, 27, 35 and 46 under 35 U.S.C. § 103(a).

II.   35 U.S.C. § 103, Alleged Obviousness Based on Owens in View of Goodwin

The Office Action rejects claims 2-3, 7-8, 13-14, 17, 21, 24, 28, 31-32, 36, 39-40 and 43-44 under 35 U.S.C. § 103(a) as being unpatentable over Owens et al. (U.S. Patent No. 6,047,284) in view of Goodwin et al. (U.S. Patent No. 6,199,195). This rejection is respectfully traversed for at least the reasons stated above with regard to independent claims 1, 12, 20, 27, 35 and 46 from which claims 2-3, 7-8, 13-14, 17, 21, 24, 28, 31-32, 36 and 39-40 depend.

With regard to claims 2-3, 7-8, 13-14 and 17, the Office Action states:

> With respect to claims 2-3 and 7-8, Owens discloses a method of deleting object data from a relational database as discussed in claim 1.
> Owens does not explicitly indicate, "wherein determining the structure of the relational database includes invoking a database meta-information class object associated with the relational database and wherein the database meta-information class object encapsulates a dependency structure of the relational database and wherein the file is an Extended Markup Language file."
> However, Goodwin discloses meta data or meta information class object as claimed (col. 12, lines 58-67 and col. 12, lines 1-19; col. 1, lines 12-67, col. 2, lines 1-50 and col. 4, lines 1-52) and a kind of XML such as UML as claimed (col. 4, lines 22-30, col. 6, lines 37-51, see fig. 3, and col. 8, lines 42-62).
> Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Owens with the teachings of Goodwin so as to obtain a method of deleting object data from a relational database because the combination would provide the method that have multiple SQL calls that may be necessary to delete an object from the relational database tables are performed as one atomic operation (col. 11, lines 37-65 and see fig. 6 and fig. 12) in the

deletion of object from an object-relational system in a customizable and database independent manner environment.

Claims 13-14 are essentially the same as claims 2-3 except that it is directed to a system rather than a method (col. 12, lines 58-67 and col. 12, lines 1-19; col. 1, lines 12-67, col. 2, lines 1-50 and col. 4, lines 1-52), and is rejected for the same reason as applied to the claims 2-3 hereinabove.

Claim 17 is essentially the same as claim 7 except that it is directed to a system rather than a method (col. 12, lines 58-67 and col. 12, lines 1-19; col. 1, lines 12-67, col. 2, lines 1-50 and col. 4, lines 1-52), and is rejected for the same reason as applied to the claim 7 hereinabove.

Owens does not teach or suggest the features of claims 1 and 12, from which claims 2-3, 7-8, 13-14 and 17 depend, respectively, as noted above. Goodwin does not provide for the deficiencies of Owens. Goodwin teaches a mechanism for automatically generating objects within extensible object frameworks and links those objects to enterprise resources. The mechanism of Goodwin involves generating logical models of source code objects, translating them into unified models, generating system definitions having templates, and generating source code objects as a function of the unified models and at least one template.

Goodwin does not teach or suggest determining a structure of the relational database, determining a delete action based on the structure of the relational database, or generating database modification commands based on the determined delete action, as recited in claim 1 and similar language found in claim 12. Since neither Owens nor Goodwin teach or suggest these features, any alleged combination of Owens and Goodwin cannot be found to teach these features. Thus, claims 2-3, 7-8, 13-14 and 17 distinguish over Owens and Goodwin at least by virtue of their dependency on claims 1 and 12 which distinguish over Owens and Goodwin.

In addition, claim 2 recites that determining the structure of the relational database includes invoking a database meta-information class object associated with the relational database. While Goodwin teaches that at least one meta data object exists in the schema repository for each application business object being modeled (column 12, lines 58-61), there is no teaching or suggestion in Goodwin to <u>determine a structure of a relational database</u> by invoking a database meta-information class object associated with the relational database. Applicant is not merely claiming metadata objects. Applicant is claiming determining the structure of a relational database, wherein such a determination

includes invoking a database meta-information class object. There is nothing in Owens or Goodwin that teaches or suggests such a feature.

With regard to claim 3, neither Owens nor Goodwin teaches or suggests that a meta-information class object encapsulates a dependency structure of the relational database. The Office Action does not even address this specific feature but merely includes its treatment of claim 3 with that of claim 2 above. Again, Applicant is not merely claiming meta data objects. Rather, Applicant is claiming in claim 3 a meta-information class object that encapsulates a dependency structure of a relational database. Such a feature is not found or suggested in either Owens or Goodwin.

As with claim 3, the Office Action merely includes claim 7 in its treatment of claim 2 and alleges that simply because Goodwin teaches meta data objects, that somehow this rises to the level of obviating the feature of a database meta-information class object being generated based on a file describing the structure and delete actions for tables in a relational database. This is not found anywhere in either Owens or Goodwin. There is not even a mention of a file that describes a structure of tables in a relational database or a file that describes delete actions for table sin a relational database. There simply is no teaching or suggestion in either reference regarding these features. Since neither reference teaches or suggests such a file as recited in claim 7, neither reference can be found to teach or suggest that such a file is an Extended Markup Language file as recited in claim 8.

Claims 13-14 and 17 are similar to claims 2, 3 and 7, respectively and are thus, allowable over the alleged combination of Owens and Goodwin for similar reasons. Therefore, in addition to being dependent on claims 1 and 12, claims 2, 3, 7, 13-14 and 17 are allowable over Owens and Goodwin based on the specific features recited therein.

With regard to claims 21, 28, 36, the Office Action's statement of the rejection is similar to that discussed above with regard to claims 2, 3, 7, 13-14 and 17 and thus, the rejection suffers from the same failings as noted above. That is, neither Owens nor Goodwin teach or suggest that generating a class object includes encapsulating information identifying a structure of a relational database and one or more delete actions. The mere teaching of meta data objects in Goodwin does not render obvious the features of claims 21, 28 and 36 which recite more than general meta data objects. To the

contrary, these claims recite encapsulating information <u>identifying a structure of a</u> <u>relational database and one or more delete actions</u>. Where is this in either Owens or Goodwin? Neither reference teaches anything having to do with encapsulating information identifying a structure of a relational database and one or more delete actions.

With regard to claims 24, 31, 39 the Office Action again relies on the general teaching in Goodwin of meta data objects as allegedly obviating these claims. However, claims 24, 31 and 39 recite that the structure of the relational database and the one or more delete actions are determined from a file describing the structure and delete actions for tables in the relational database. This is much more than general meta data objects and the mere teaching of meta data objects does not render the features of claims 24, 31 and 39 obvious.

Regarding claims 32 and 40, the Office Action states that Goodwin teaches JDBC and thus, the features of claims 32 and 40 are rendered obvious when Goodwin is combined with Owens. Even though Goodwin teaches JDBC, there is nothing in Goodwin that teaches the use of JDBC <u>database metadata</u> associated with <u>a relational</u> <u>database</u> to <u>generate a file</u> describing the <u>structure and delete actions of tables in the</u> <u>relational database</u>. The Office Action appears to be picking out terms in the claims, such as JDBC, finding those terms and merely asserting that everything else is obvious regardless of the context and other features present in the claims. Merely teaching JDBC does not mean that the use of JDBC database metadata associated with a relational database to generate a file describing the structure and delete actions of tables in a relational database is taught or suggested. Applicant respectfully submits that the Office Action has not taken into account all of the features of claims 32 and 40 and has merely zeroed in on JDBC without regard for the actual features recited in claims 32 and 40.

With regard to claims 43-44, the Office Action again relies entirely on the general teaching of a meta data object in Goodwin to allegedly obviate the features of claims 43 and 44. However, claims 43 and 44 recite more than simply a meta data object. Claim 43 specifically recites "a meta-information class for determining a structure of the relational database and one or more delete actions based on the structure of the relational database; and a database meta-information generator class for generating the class object

based on the determined structure and the determined one or more delete actions." There is nothing in the general teaching of a meta data object that teaches or suggests the very specific meta-information class and meta-information generator class recited in claims 43-44. Again, the Office Action has merely jumped on a term, meta data or meta information, and once that term was found in a reference, alleged that the reference teaches everything else in the claim. This clearly is not the case. Neither Owens nor Goodwin have anything to do with determining a structure of a relational database or delete actions. Therefore, they cannot teach or suggest a meta-information class that determines such a structure or delete action, let alone a generator class for generating a class object based on a determined structure and one or more delete actions.

In view of the above, Applicant respectfully submits that neither Owens nor Goodwin, either alone or in combination, teach or suggest the features of claims 2-3, 7-8, 13-14, 17, 21, 24, 28, 31-32, 36- 39-40 and 43-44. Accordingly, Applicant respectfully requests withdrawal of the rejection of claims 2-3, 7-8, 13-14, 17, 21, 24, 28, 31-32, 36-39-40 and 43-44 under 35 U.S.C. § 103(a).

III.   **35 U.S.C. § 103, Alleged Obviousness Based on Owens and Goodwin in View of Crus**

The Office Action rejects claims 4-6, 9-10, 15-16, 18-19, 22-23, 25-26, 29-30, 33-34, 37-38, 41-42 and 45 under 35 U.S.C. § 103(a) as being unpatentable over Owens et al. in view of Goodwin et al. and further in view of Crus et al. (U.S. Patent No. 4,947,320). This rejection is respectfully traversed.

The rejections of these claims are based on the alleged combination of Owens and Goodwin discussed above and thus, the rejection suffers from the same failings as noted above. The rejection of these claims includes the addition of the reference Crus which allegedly teaches delete action identifiers such as cascade delete and nullify columns delete at column 1, lines 62-67, column 2, lines 1-61, column 5, lines 1-67, column 6, lines 1-36, column 16, lines 60-67, column 17, lines 1-67 and column 18, lines 1-18. In actuality, all that Crus teaches is enforcing referential constraints in a database management system (column 1-2), consulting a table descriptor to determine if there are

any referential constraints to be enforced for a specific operation (column 5-6), and cascade and nullify columns delete rules (column 16-18). Applicant has not claimed to have invented enforcing referential constraints or the cascade and nullify columns delete operations.

Applicant claims a database meta-information class object that includes a delete action identifier for each dependent table of a plurality of tables in a relational database (claims 4 and 15), a delete action that is determined for a relational database or a deleted action identifier being one of cascade delete and nullify columns delete (claims 5, 6, 16, 22, 23, 29, 30, 37, 38 and 45), user input to override default delete action identifiers in a file (claim 9, 25, 33, and 41), user input to insert one or more delete constraints in a file for one or more of the tables of the relational database (claims 10, 26, 34 and 42), a file editor application that changes delete actions in a file fro one or more tables in the relational database based on user input to override default delete action identifiers in the file (claim 18), the editor application inserts one or more delete constraints into the file for one or more of the tables in the relational database based on user input (claim 19). None of these features are taught or suggested by the mere teaching in Crus of enforcing referential constraints and that a delete rule may be a cascade delete rule or a nullify columns delete rule. In fact, the features of meta-information class objects that include delete action identifiers, user input to override default delete actions or to insert one or more delete constraints in a file and a file editor are not even addressed by the Office Action.

In view of the above, Applicant respectfully submits that neither Owens, Goodwin nor Crus, either alone or in combination, teach or suggest the features of claims 4-6, 9-10, 15-16, 18-19, 22-23, 25-26, 29-30, 33-34, 37-38, 41-42 and 45. Accordingly, Applicant respectfully requests withdrawal of the rejection of claims 4-6, 9-10, 15-16, 18-19, 22-23, 25-26, 29-30, 33-34, 37-38, 41-42 and 45 under 35 U.S.C. § 103(a).

IV. 35 U.S.C. § 103, Alleged Obviousness Based on Owens in View of Crus

The Office Action rejects claims 47-48 under 35 U.S.C. § 103(a) as being unpatentable over Owens et al. in view of Crus et al.. This rejection is respectfully traversed.

As to claims 47-48, the Office Action states in pertinent part:

> Owens does not explicitly indicate, "wherein the user input overrides one or more of the one or more default delete actions and wherein the user input inserts one or more delete action constraints."
> However, Crus discloses delete action identifier such as cascade delete, delete set null as well as nullity columns delete and delete constraints as claimed (abstract, col. 1, lines 62-67, col. 2, lines 1-61, col. 5, lines 1-67, col. 6, lines 1-36, col. 16, lines 60-67, col. 17, lines 1-67 and col. 18, lines 1-18)...(emphasis added)

The faults with the combination of Owens and Crus have been previously discussed above. Despite the allegations made by the Office Action, Crus in fact does not teach or suggest that a user input overrides one or more default delete action and inserts one or more delete action constraints. Crus makes no mention whatsoever regarding user input for overriding default delete actions. The mere teaching of delete actions such as cascade delete and nullify columns delete in no way obviates a feature of user input overriding one or more default delete actions. Moreover, the mere teaching of enforcing referential constraints does not obviate a user input inserting one or more delete action constraints, as recited in claims 47 and 48. The sections cited by the Examiner do not even mention user input, let alone user input overriding one or more default delete actions or inserting delete action constraints.
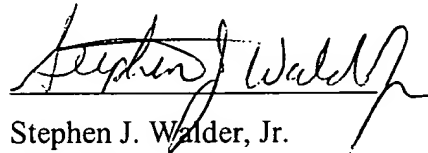
In view of the above, Applicant respectfully submits that neither Owens nor Crus, either alone or in combination teach or suggest the features of claims 47 and 48. Accordingly, Applicant respectfully requests withdrawal of the rejection of claims 47-48 under 35 U.S.C. § 103(a).

V. Conclusion

It is respectfully urged that the subject application is patentable over Owens, Goodwin and Crus and is now in condition for allowance. The Examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the Examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

Respectfully submitted,

DATE: August 22, 2002

Stephen J. Walder, Jr.
Reg. No. 41,534
Carstens, Yee & Cahoon, LLP
P.O. Box 802334
Dallas, TX 75380
(972) 367-2001
Attorney for Applicant